

Hi all!

Before we start with the manual, we would like to introduce our selves. We are just regular people like you who are fighting with diabetes. We are not a huge corporation, which has developed something for money, and we also don't claim that we have created the most genius thing in the world. We have gathered together and each one of us contributed to the project with his own skills.

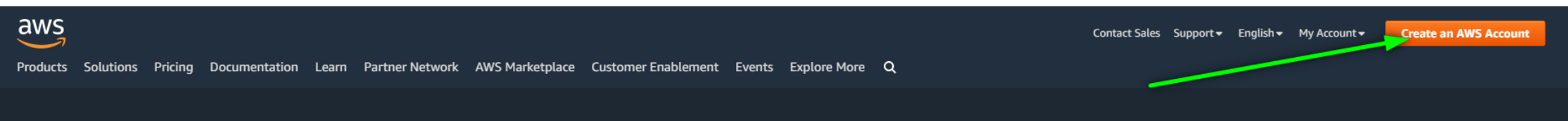
The developer of the code's name is Yaron Khazai, who is a CTO at a software company which I – George Mustakou also used to work for. The design touch has been provided by Ivan Ivanou from Don't Blink Studio, who is a very good friend of the diabetes community.

If anyone would like to contribute to the project with anything (code, UI, etc), please feel free to contact Yaron through his github.

1. Please follow the link:

<https://aws.amazon.com/>

2. Select 'Create and AWS Account' at the top right corner:



3. The first step of the registration process will open. Just follow the instructions on the screen.



Explore Free Tier products with a new AWS account.

To learn more, visit aws.amazon.com/free.



Sign up for AWS

Email address

You will use this email address to sign in to your new AWS account.

⚠ An email address is required.

Password

Confirm password

AWS account name

Choose a name for your account. You can change this name in your account settings after you sign up.

Continue (step 1 of 5)

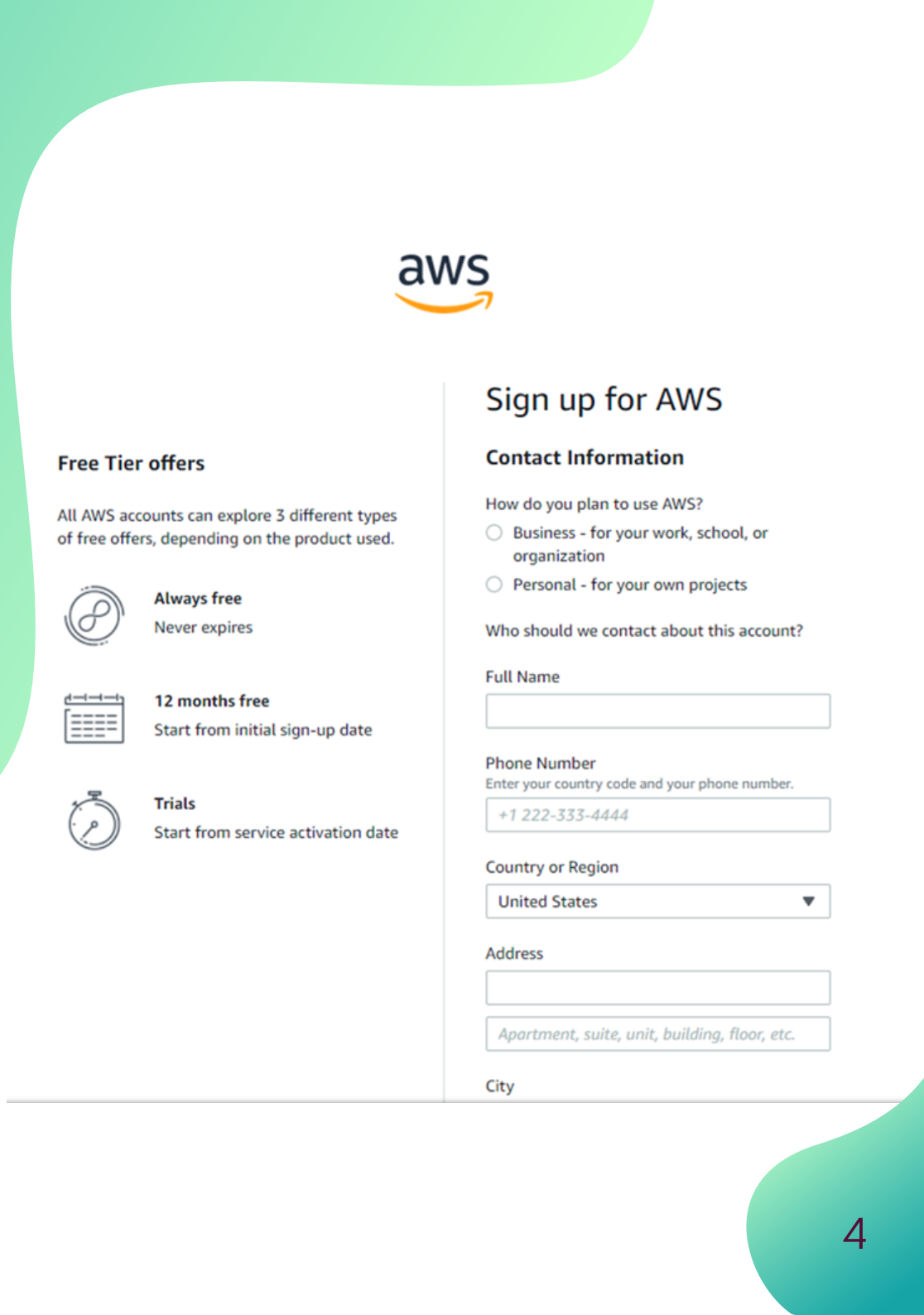
[Sign in to an existing AWS account](#)

3.1. In step 1 enter your e-mail address, which you will also use as username to log in.

Create a strong password and confirm it in the next field.

Choose a name for your account. It can be changed later from the Settings.

Pay attention to the format (given in light grey) of the phone number, which is requested, or the validation will not let you continue to the next step.





Secure verification

i We will not charge for usage below AWS Free Tier limits. We temporarily hold \$1 USD/EUR as a pending transaction for 3-5 days to verify your identity.



Sign up for AWS

Billing Information

Credit or Debit card number

⚠ The credit card number is required.



AWS accepts all major credit and debit cards. To learn more about payment options, review our [FAQ](#)

Expiration date

Month

▼

Year

▼

Cardholder's name

Billing address

☒ Use my contact address



☐ Use a new address

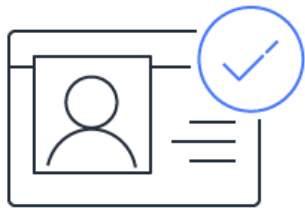
Verify and Continue (step 3 of 5)

You might be redirected to your bank's website to authorize the verification charge.

3.3. In step 3 enter valid card details.

Amazon will block 1 USD/EUR from your account and give it back to you after 3-5 days. This is just to verify your identity.

The service which we are going to use is free of charge



Sign up for AWS

Confirm your identity

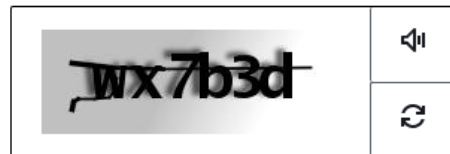
Before you can use your AWS account, you must verify your phone number. When you continue, the AWS automated system will contact you with a verification code.

Country or region code

United States (+1)

Mobile phone number

Security check



Type the characters as shown above

Send SMS (step 4 of 5)

3.4. Step 4 is asking for your phone number, so that AWS could send you an SMS with a verification code.

In case you don't receive an SMS, contact Amazon support and they will verify the account for you.



Sign up for AWS

Select a support plan

Choose a support plan for your business or personal account. [Compare plans and pricing examples](#)

[🔗](#) You can change your plan anytime in the AWS Management Console.

☒ Basic support - Free

- Recommended for new users just getting started with AWS
- 24x7 self-service access to AWS resources
- For account and billing issues only
- Access to Personal Health Dashboard & Trusted Advisor



☐ Developer support - From \$29/month

- Recommended for developers experimenting with AWS
- Email access to AWS Support during business hours
- 12 (business)-hour response times



☐ Business support - From \$100/month

- Recommended for running production workloads on AWS
- 24x7 tech support via email, phone, and chat
- 1-hour response times
- Full set of Trusted Advisor best-practice recommendations



Need Enterprise level support?

From \$15,000 a month you will receive 15-minute response times and concierge-style experience with an assigned Technical Account Manager. [Learn more](#) [🔗](#)

Complete sign up

3.5. At step 5 select 'Basic support - Free' plan and then click on 'Complete sign up'



Congratulations!

Thank you for signing up with AWS.

We are activating your account, which should take a few minutes. You will receive an email when this is complete.

[Go to the AWS Management Console](#)

[Sign up for another account](#) or [Contact Sales](#)

3.6. On the 'Congratulations' page, click on the large yellow button 'Go to the AWS Management Console'.



Sign in

1

☒ **Root user**

Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☐ **IAM user**

User within an account that performs daily tasks. [Learn more](#)

Root user email address

2

3

Next

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

————— New to AWS? —————

Create a new AWS account


3.7. Sign in by selecting 'Root User', enter your e-mail address and click 'Next'



Security check

For security reasons, we need to verify that account holders are real people.

Type the characters seen in the image below

A security check image showing the characters 'dbzd4g' in a distorted, pixelated font. The characters are overlaid with a grid of lines. To the right of the image are two buttons: a speaker icon for audio and a refresh icon.

dbzd4g

1

2

3.8. Pass the security check by entering the characters in the image and click 'Submit'



Root user sign in ⓘ

Email: [redacted]@gmail.com

Password

[Forgot password?](#)

1

2

Sign in

[Sign in to a different account](#)

[Create a new AWS account](#)

3.9. Now enter your password and click 'Sign in'

4. Now you've taken to the initial page of the console.

AWS Management Console

AWS services


► All services

Build a solution

Get started with simple wizards and automated workflows.


Launch a virtual machine

With EC2
2-3 minutes




Build a web app

With Elastic Beanstalk
6 minutes




Build using virtual servers

With Lightsail
1-2 minutes




Register a domain

With Route 53
3 minutes




Connect an IoT device

With AWS IoT
5 minutes




Start migrating to AWS

With AWS MGN
1-2 minutes




Start a development project

With CodeStar
5 minutes




Deploy a serverless microservice

With Lambda, API Gateway
2 minutes



► See more

Stay connected to your AWS resources on-the-go

 AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device. [Learn more](#)

Explore AWS

Free AWS Training

Advance your career with AWS Cloud Practitioner Essentials—a free, six-hour, foundational course. [Learn more](#)

Modernize Your APIs with GraphQL

AWS AppSync is a fully-managed GraphQL service that improves app performance and developer productivity. [Learn more](#)

Free AWS Training

Complete projects faster and troubleshoot with confidence with 500+ free digital courses covering AWS products and services. [Learn more](#)

Join the Graviton Challenge

Move your workloads to Graviton2 and save

4.1. In the search field start typing 'Lambda' and click on the result which appears.

The screenshot shows the AWS Management Console interface. At the top, the search bar is highlighted with a blue circle and the number '1'. Below the search bar, the search results for 'Lambda' are displayed. The 'Lambda' service is highlighted with a blue circle and the number '2'. The interface includes a sidebar on the left with navigation links like 'AWS services', 'Build a solution', 'Launch a virtual machine', and 'Connect an IoT device'. The main content area shows search results for 'Lambda', including a list of services (Lambda, CodeBuild, AWS Signer, Amazon Lex) and features (Local processing, Target groups). The right sidebar contains promotional content for AWS services and training.

1

Search results for 'lam'

2

Lambda
Run Code without Thinking about Servers

CodeBuild
Build and Test Code

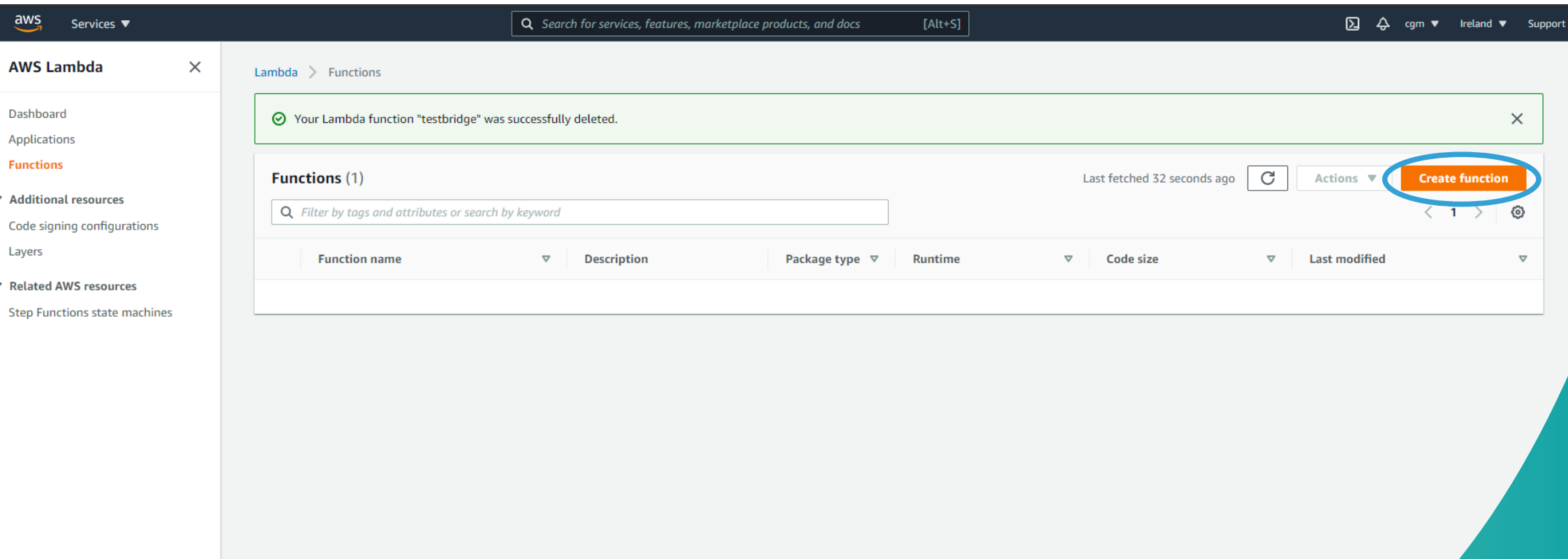
AWS Signer
Ensuring trust and integrity of your code

Amazon Lex
Build Voice and Text Chatbots

Local processing
IoT Core feature

Target groups
EC2 feature

4.2. You will be transferred to the screen below. Click on the orange button 'Create Function' in the right corner.



4.3. In the 'Function name' field enter a name. It can be anything. From the dropdown menu select Python 3.6 or 3.7. Any will do. Click on the 'Create Function' button at the bottom right corner.

aws Services ▾

Search for services, features, marketplace products, and docs [Alt+S]

cgml Ireland ▾ Support ▾

Lambda > Functions > Create function

Create function [Info](#)

Choose one of the following options to create your function.

Author from scratch ☒

Start with a simple Hello World example.

Use a blueprint ☐

Build a Lambda application from sample code and configuration presets for common use cases.

Container image ☐

Select a container image to deploy for your function.

Browse serverless app repository ☐

Deploy a sample Lambda application from the AWS Serverless Application Repository.

Basic information

Function name
Enter a name that describes the purpose of your function.

Name

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime [Info](#)
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Python 3.6

Permissions [Info](#)

By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

► Change default execution role

► Advanced settings

Cancel **Create function**

4.4. You will see the screen below:

The screenshot displays the AWS Lambda console interface. At the top, a green notification bar states: "Successfully created the function Name. You can now change its code and configuration. To invoke your function with a test event, choose 'Test'".

The main content area is titled "Name" and includes a breadcrumb trail: "Lambda > Functions > Name". On the right side of the header, there are buttons for "Throttle", "Copy ARN", and "Actions".

The "Function overview" section is expanded, showing the following details:

- Name:** Name
- Layers:** (0)
- Description:** -
- Last modified:** 26 seconds ago
- Function ARN:** arn:aws:lambda:eu-west-1:191413258258:function:Name

Below the overview, there are two buttons: "+ Add trigger" and "+ Add destination".

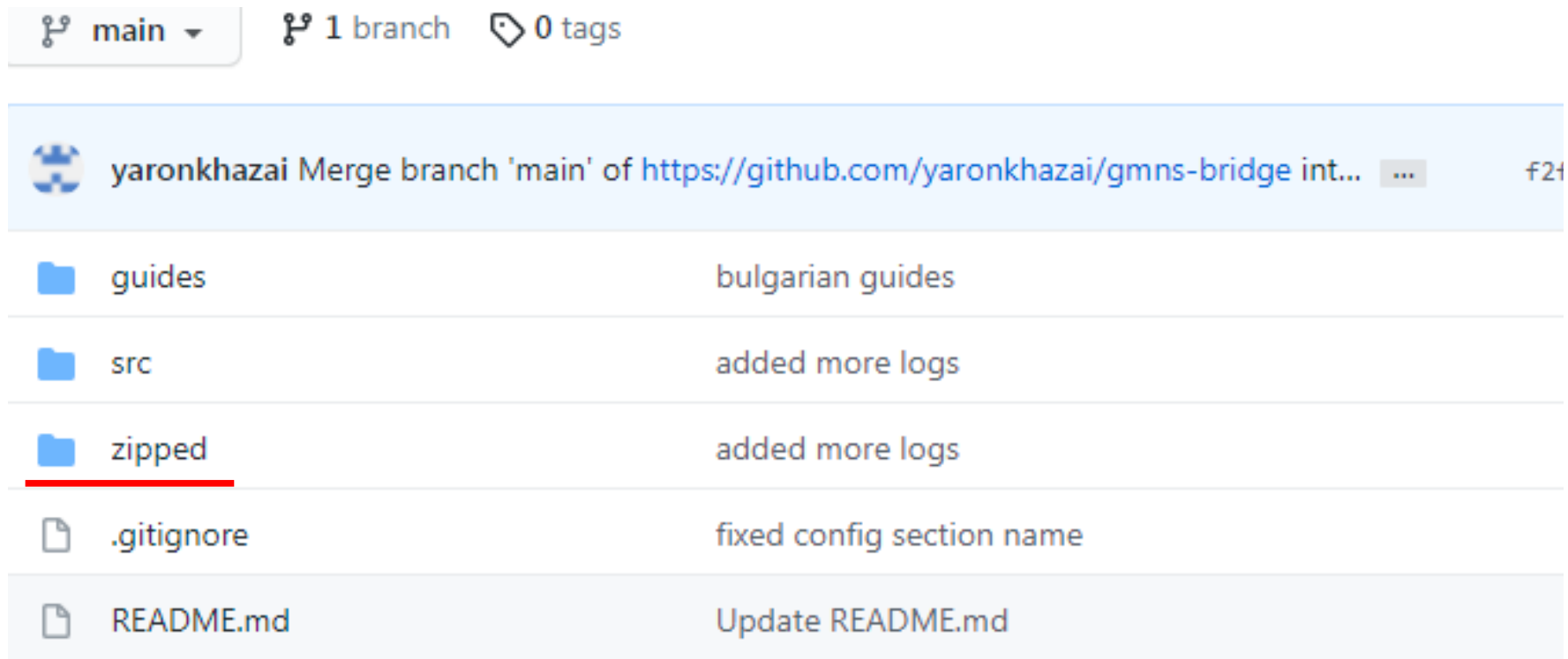
The console has a tabbed interface with the following tabs: "Code" (selected), "Test", "Monitor", "Configuration", "Aliases", and "Versions".

The "Code source" section is visible, showing a file explorer with a folder named "Name" containing a file named "lambda_function.py". The "Code" tab is active, and the "Test" button is highlighted in orange. A green status bar at the bottom of the code editor indicates "Changes deployed".

5. Now follow the link:

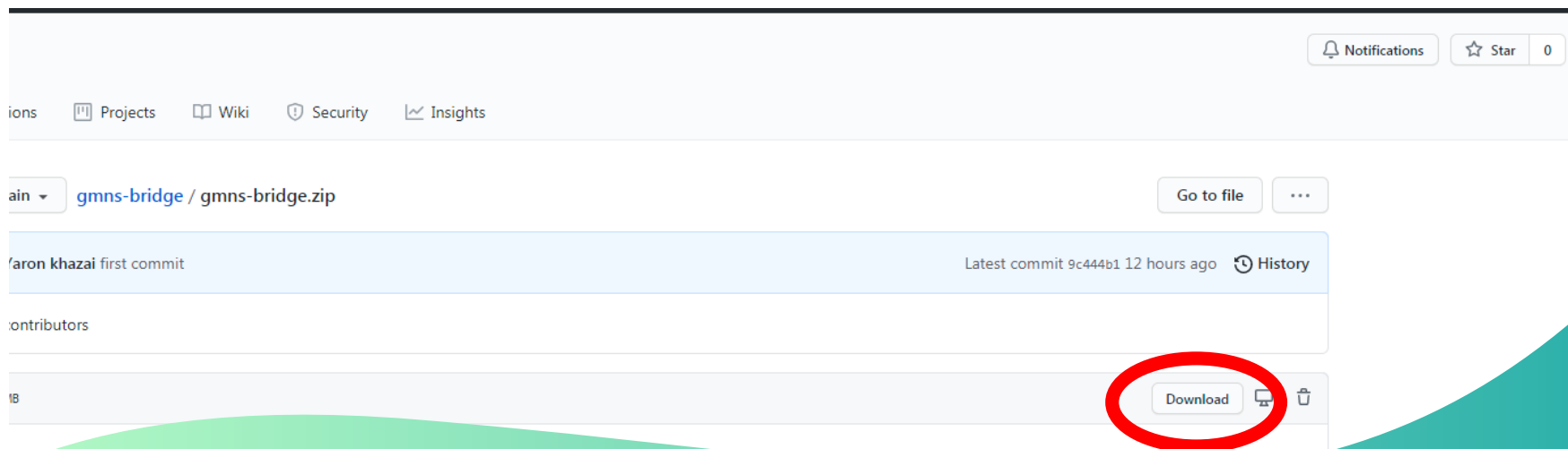
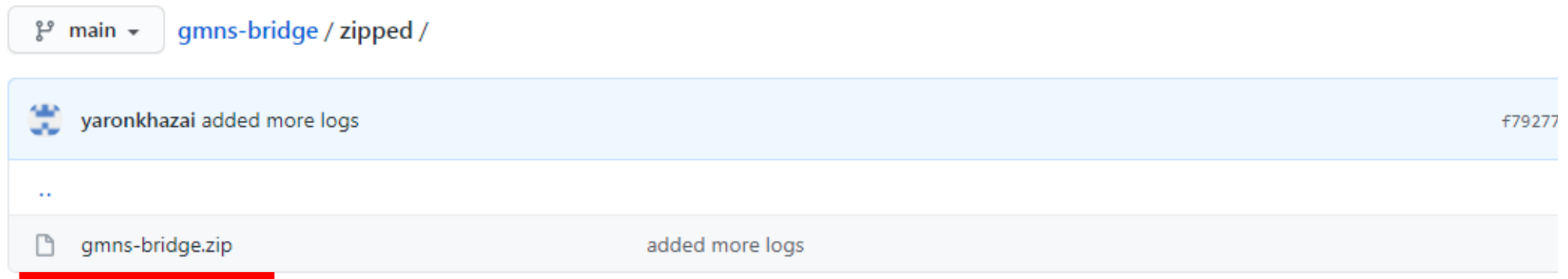
<https://github.com/yaronkhazai/gmns-bridge>

5.1. Locate the folder 'Zipped' and double-click on it to open it.



5.2. Click on the gmns-bridge.zip

5.3. Now click on the 'Download' button and remember the location of the file (desktop, Downloads folder, wherever you download stuff)



6. Go back to the function and from the dropdown 'Upload from' select .zip file

Services ▾ Search for services, features, marketplace products, and docs [Alt+S] cgm ▾ Ireland ▾


✔ Successfully created the function Name. You can now change its code and configuration. To invoke your function with a test event, choose "Test".


Lambda > Functions > Name

Name

Throttle Copy ARN Actions ▾

▼ Function overview Info

 Name


 Layers (0)

+ Add trigger

+ Add destination

Description
-

Last modified
2 minutes ago

Function ARN
 arn:aws:lambda:eu-west-1:191413258258:function:Name

Code Test Monitor Configuration Aliases Versions

Code source Info

File Edit Find View Go Tools Window Test ▾ Deploy Changes deployed

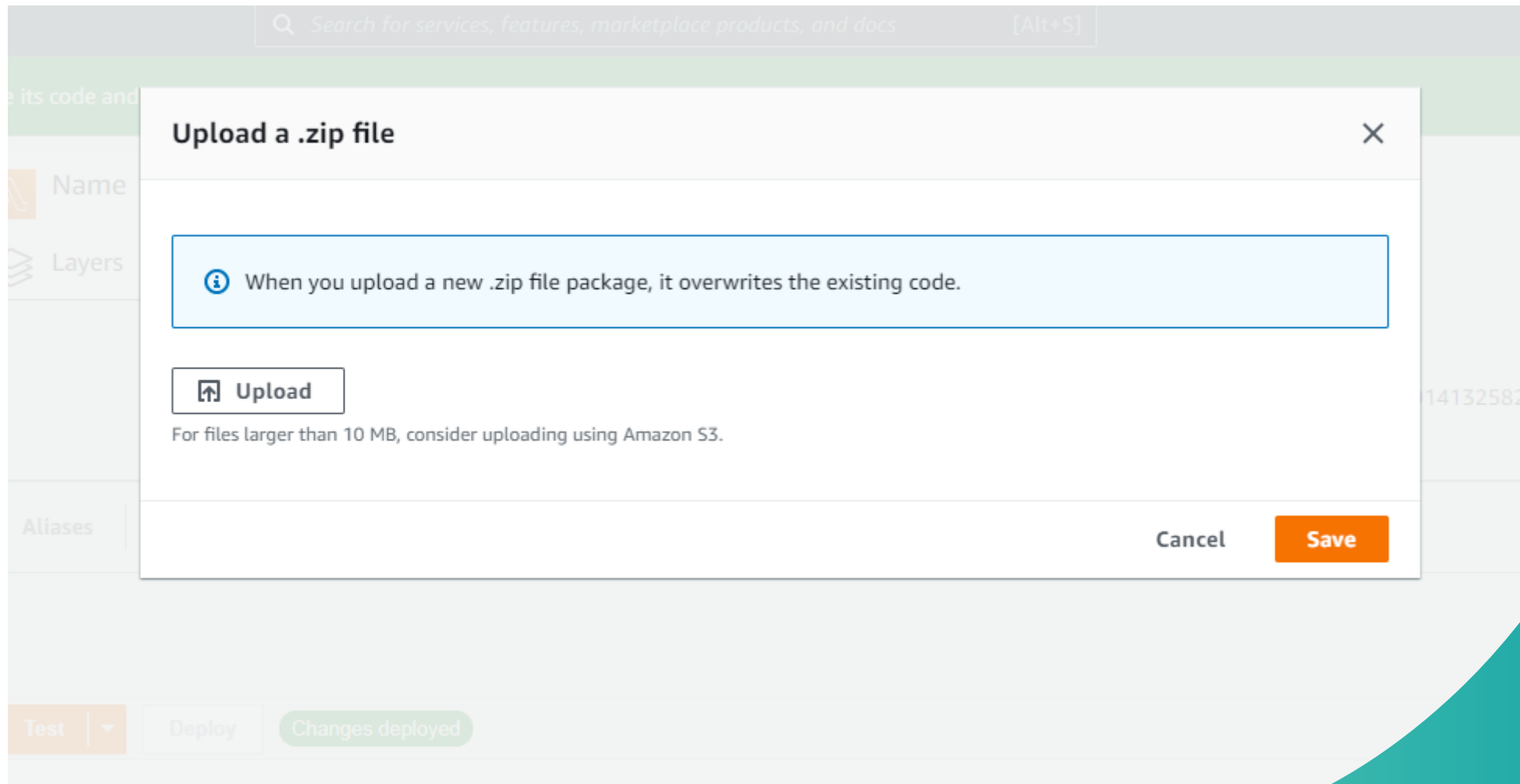
Go to Anything (Ctrl-P)

Environment

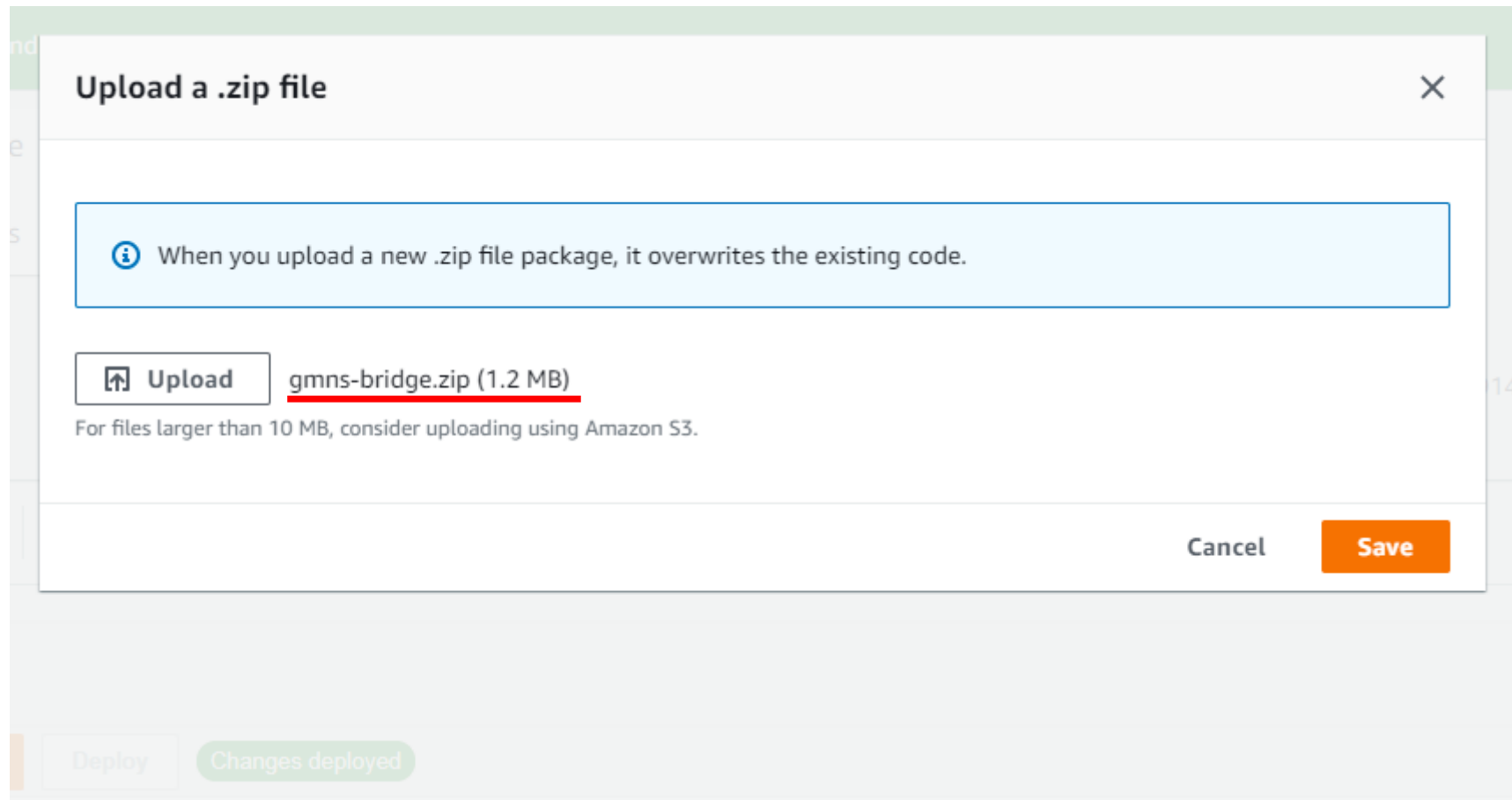
- Name - /
 - lambda_function.py

Upload from ▲
- .zip file
- Amazon S3 location

6.1. A window opens which will help us upload our file. Click on 'Upload' and select the zip file which we have just downloaded.



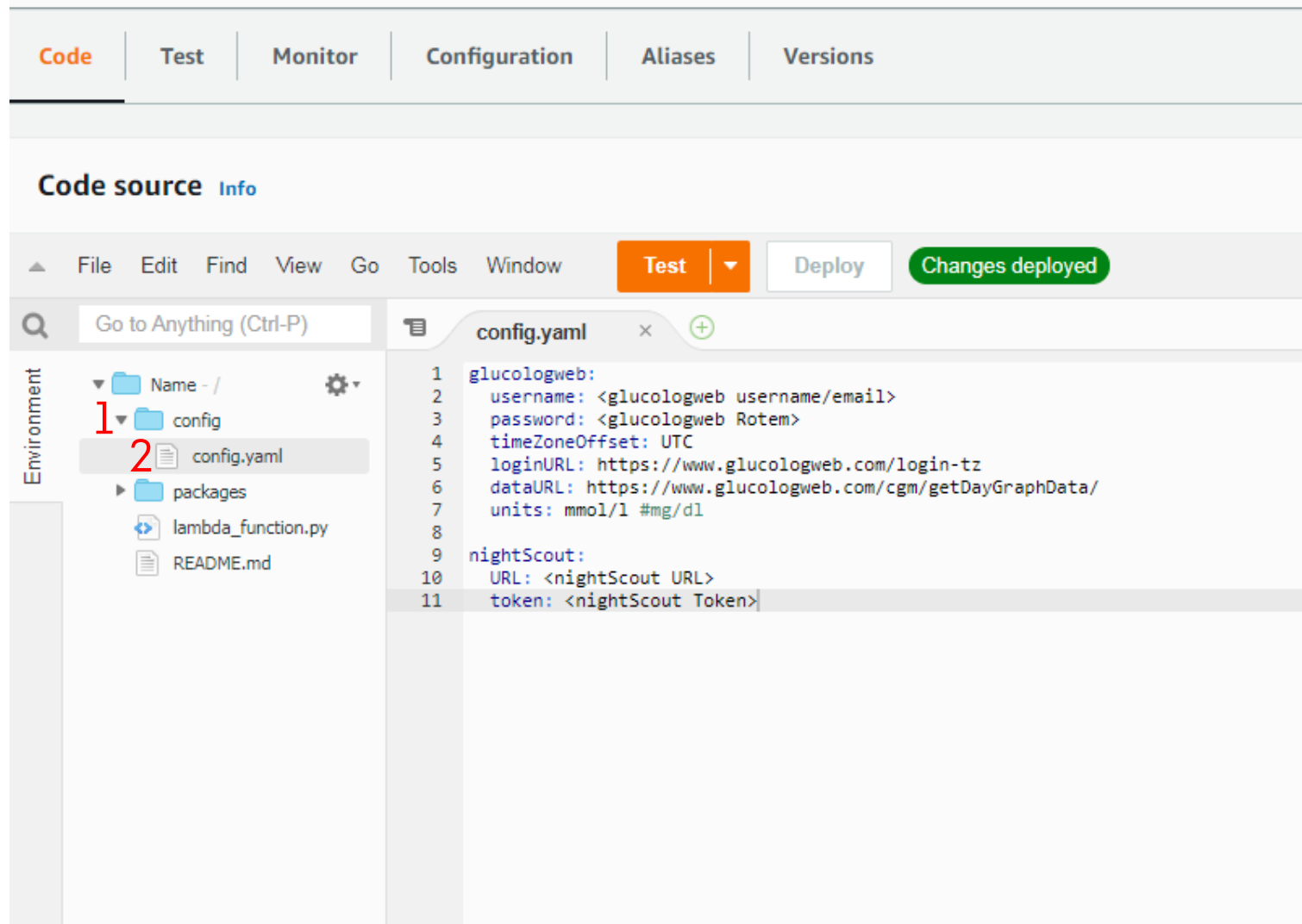
6.2. Once the file is uploaded, the window will confirm it. Click the 'Save' button



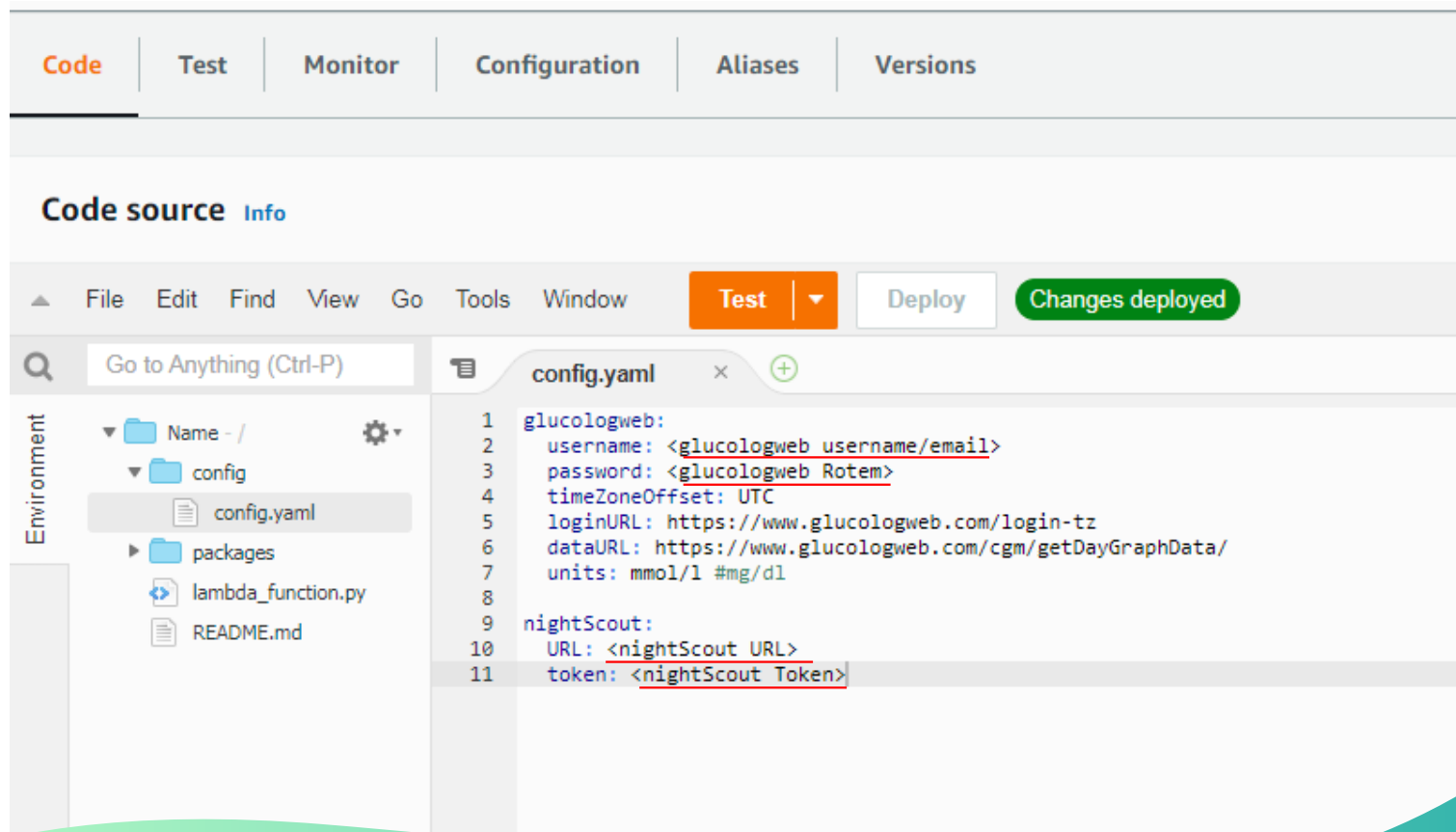
6.3. Back to the function. If everything was uploaded successfully, you will see that now there are a few folders on the left side.

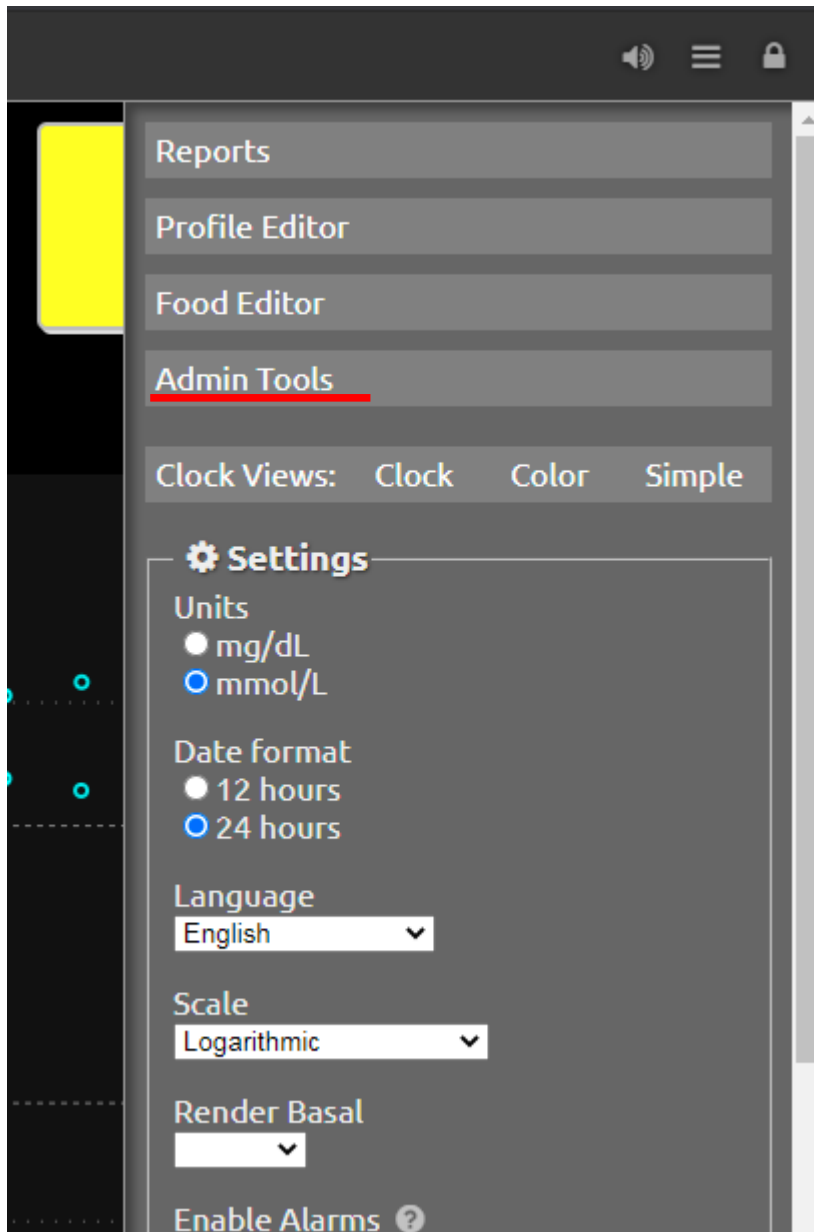
The screenshot displays the AWS Lambda console interface. At the top, a green notification bar states "Successfully updated the function Name." Below this, the function details for "Name" are shown, including its description, last modified time (5 minutes ago), and Function ARN. The "Code" tab is selected, and the "Code source" section is visible. On the left side of the code editor, a file explorer is open, showing the project structure: "Name" (folder), "config" (folder), "packages" (folder), "lambda_function.py" (file), and "README.md" (file). A blue circle highlights the file explorer. The "Test" button is highlighted in orange, and a "Changes deployed" status is shown. The bottom right corner features a large, stylized "A" logo.

6.4. Click on the folder called 'Config' and then double-click on the config.yaml file to open the configurator.



6.5. Replace the username and password, which are under the 'glucologweb' category. These are you login credentials to glucologweb.com. Replace also the URL with you Nightscout URL and token parameters.





7. The last parameter 'token' is in your nightscout Admin Tools. If you don't have one, here is how to generate it:

7.1. Load your nightscout and from the 3 lines at the top right click on 'Admin Tools'

7.2. Click on 'Add new subject'

Admin Tools

Subjects - People, Devices, etc

Each subject will have a unique access token and 1 or more roles. Click on the access token to open a new view with

Name	Roles	Access Token	Notes
<div><div>Add new Subject</div><div>Database contains 0 subjects</div></div>			

Roles - Groups of People, Devices, etc

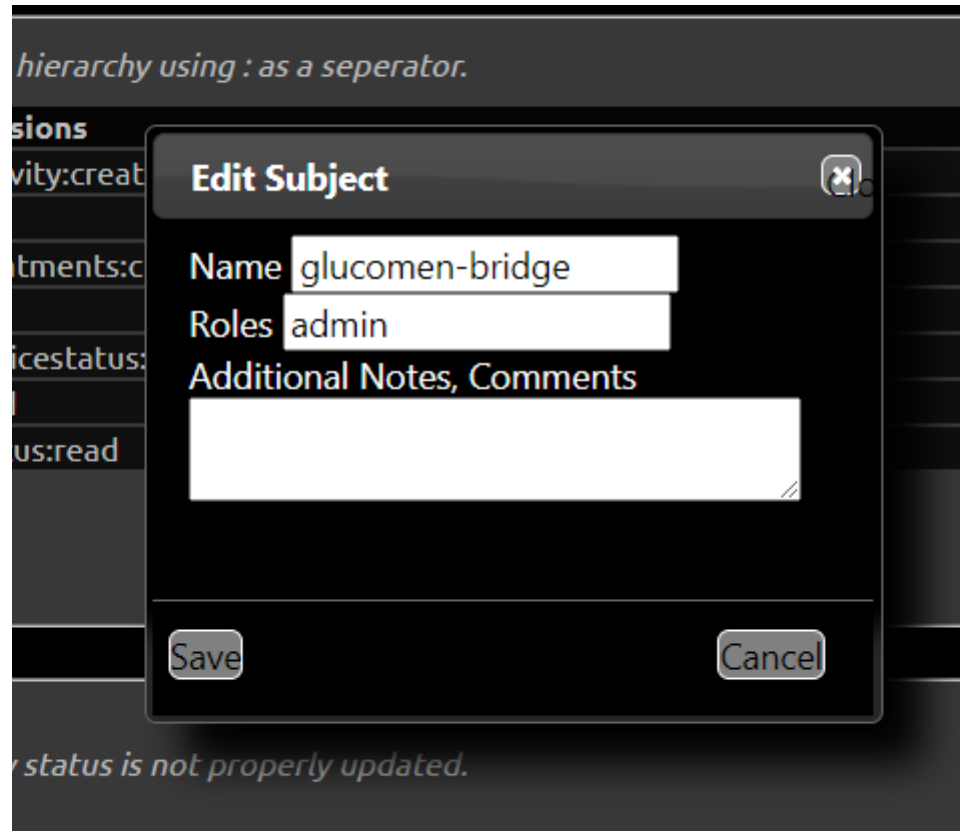
Each role will have a 1 or more permissions. The * permission is a wildcard, permissions are a hierarchy using : as a s

Name	Permissions
✓ activity	api:activity:create
✓ admin	*
✓ careportal	api:treatments:create
✓ denied	[none]
✓ devicestatus-upload	api:devicestatus:create
✓ readable	*.*:read
✓ status-only	api:status:read

Add new Role

Database contains 7 roles

7.3. Give it a name (it can be anything) and give an 'admin' role. Click 'Save'





The screenshot shows a dark-themed software interface with a modal dialog box titled "Edit Subject". The dialog has a close button (X) in the top right corner. It contains three input fields: "Name" with the text "glucomen-bridge", "Roles" with the text "admin", and "Additional Notes, Comments" which is an empty text area. At the bottom of the dialog are two buttons: "Save" and "Cancel". In the background, a list of permissions is visible, including "vity:creat", "tments:c", "icestatus:", "us:read", and "status is not properly updated."

7.4. Your token is now visible and it contains parts from the name you gave it and digits.

Admin Tools

Subjects - People, Devices, etc






Each subject will have a unique access token and 1 or more roles. Click on the access token to open a new view with the selected subject, this secret link can then be shared.

Name	Roles	Access Token	
  glucomen-bridge	admin	glucomenbr-60	

[Add new Subject](#) Database contains 1 subjects

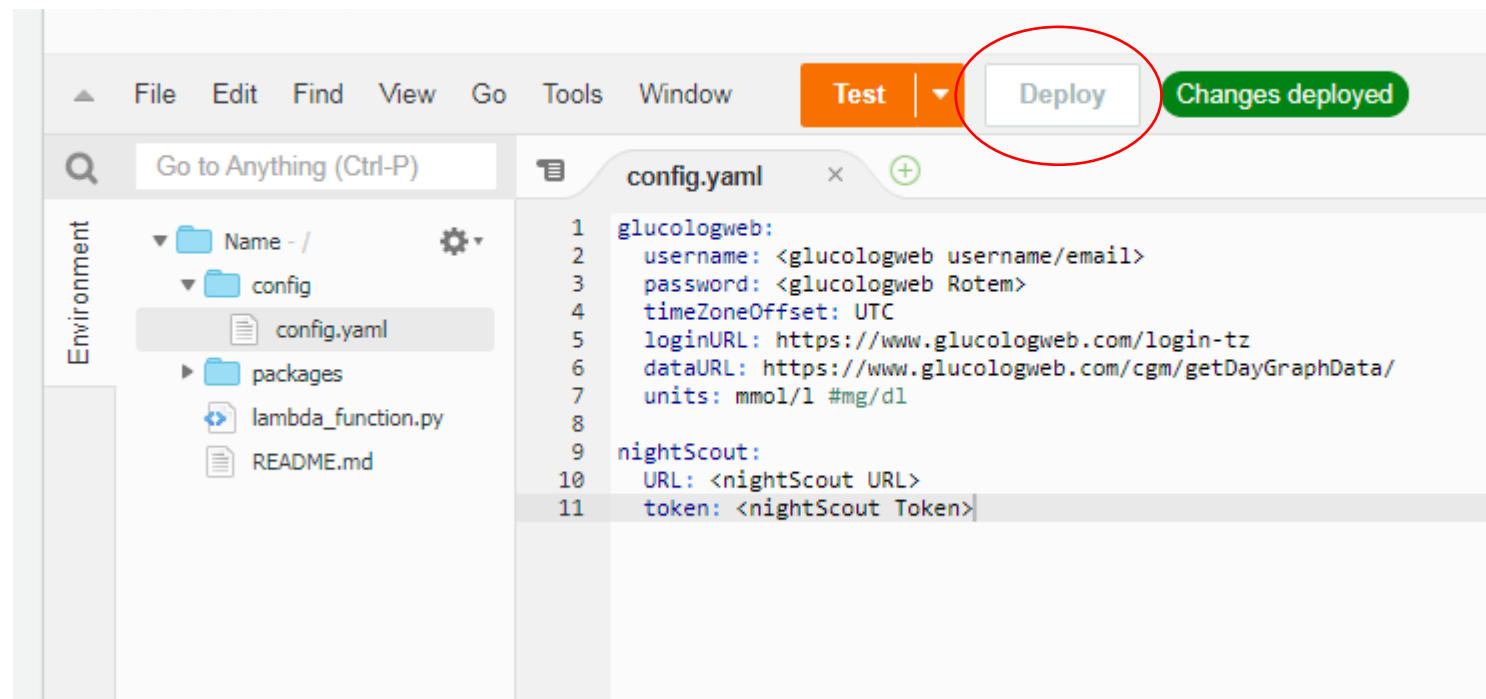
Roles - Groups of People, Devices, etc

Each role will have a 1 or more permissions. The * permission is a wildcard, permissions are a hierarchy using : as a separator.

Name	Permissions	Notes
 activity	api:activity:create	[system default]
 admin	*	[system default]
 careportal	api:treatments:create	[system default]
 denied	[none]	[system default]
 devicestatus-upload	api:devicestatus:create	[system default]

7.5. Now you are ready to save the changes you've made to the config file by clicking the 'Deploy' button. It functions like a 'Save' button. Every time when you change anything from the config, you have to click on the 'Deploy' button in order for the changes to take effect.

That's it! Your function is ready! What's left is to configure something which will trigger our function to be executed automatically as per scheduled time.

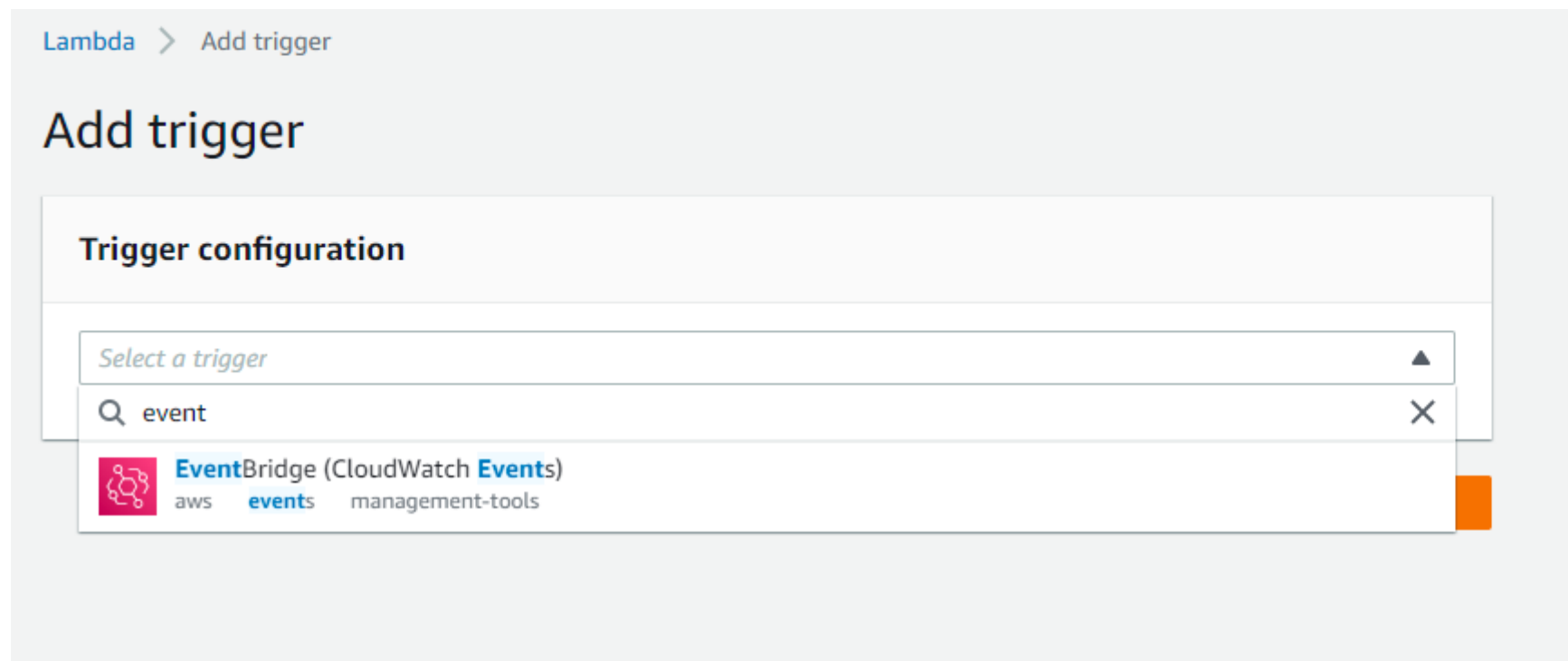


8. Click on 'Add trigger'


The screenshot shows the AWS Lambda console interface. At the top, the 'function overview' section displays the function name and a list of layers (currently 0). A red circle highlights the '+ Add trigger' button. Below this, a tabbed interface shows 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. The 'Code' tab is active, showing the 'Code source' section. A file explorer on the left shows the project structure, including 'Name - /', 'config', 'packages', 'lambda_function.py', and 'README.md'. The 'config.yaml' file is selected, and its contents are displayed in the editor:

```
1 glucologweb:
2   username: <glucologweb username/email>
3   password: <glucologweb Rotem>
4   timeZoneOffset: UTC
5   loginURL: https://www.glucologweb.com/login-tz
6   dataURL: https://www.glucologweb.com/cgm/getDayGraphData/
7   units: mmol/l #mg/dl
8
9 nightScout:
10  URL: <nightScout URL>
```

8.1. In the search start typing 'event' and click on the result 'Event Bridge'



Trigger configuration

 **EventBridge (CloudWatch Events)**
aws events management-tools

Rule
Pick an existing rule, or create a new one.

☒ Create a new rule

☐ Existing rules

Rule name*
Enter a name to uniquely identify your rule.

Rule description
Provide an optional description for your rule.

Rule type
Trigger your target based on an event pattern, or based on an automated schedule.


☐ Event pattern

☒ Schedule expression

Schedule expression*
Self-trigger your target on an automated schedule using Cron or rate expressions. Cron expressions are in UTC.

e.g. rate(1 day), cron(0 17 ? * MON-FRI *)

Lambda will add the necessary permissions for Amazon EventBridge (CloudWatch Events) to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

 The Lambda console no longer supports disabling EventBridge (CloudWatch Events) triggers. Delete these triggers to stop further actions.

Cancel Add

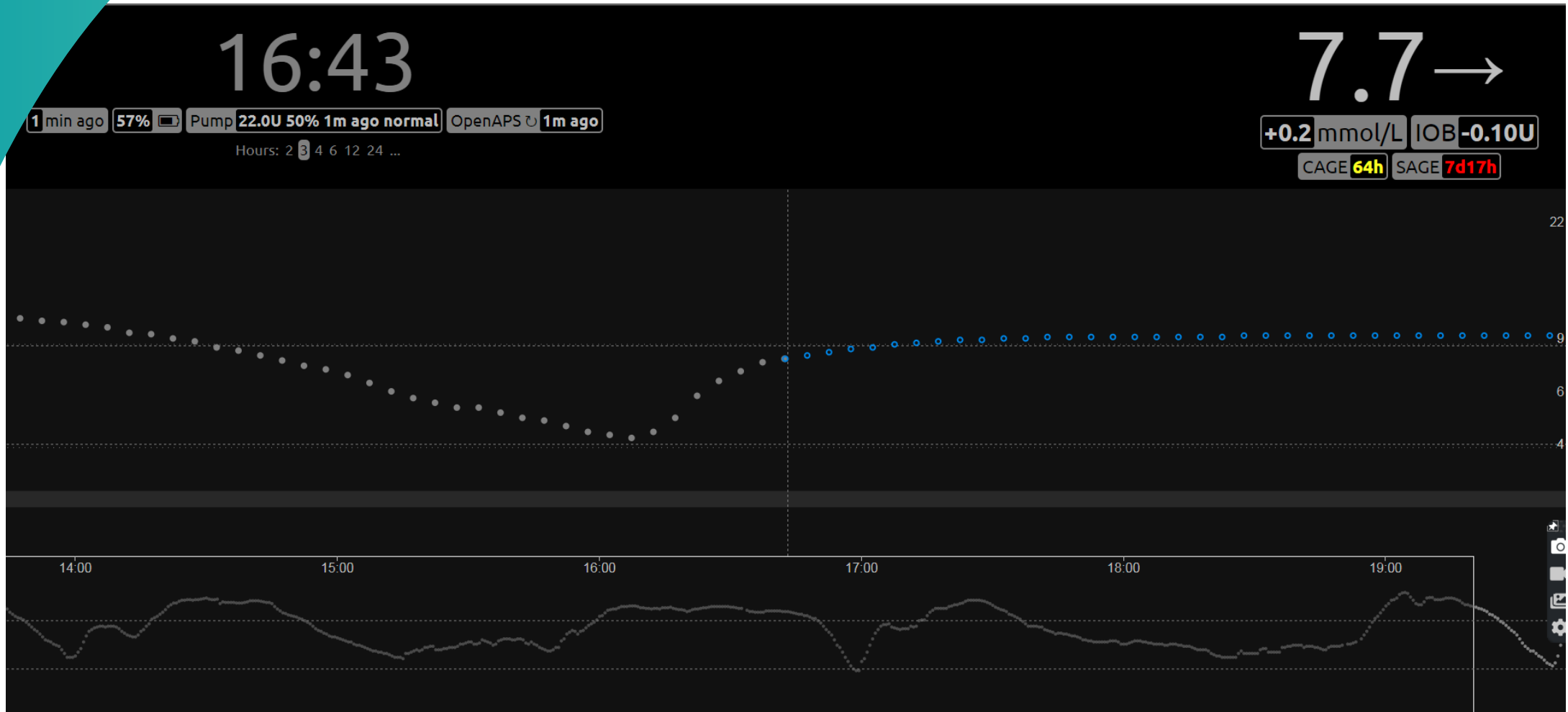
8.2. In the Trigger Configuration:

- Select 'Create new rule'
- Under 'Rule name' enter a name. It can be anything
 - Under 'Rule description' type in every_1_minute
 - Under 'Rule type' select 'Schedule expression'

Under 'Schedule Expression*' type in EXACTLY rate(1 minute). The expression is case sensitive, so be careful. There is space only between 1 and minute.

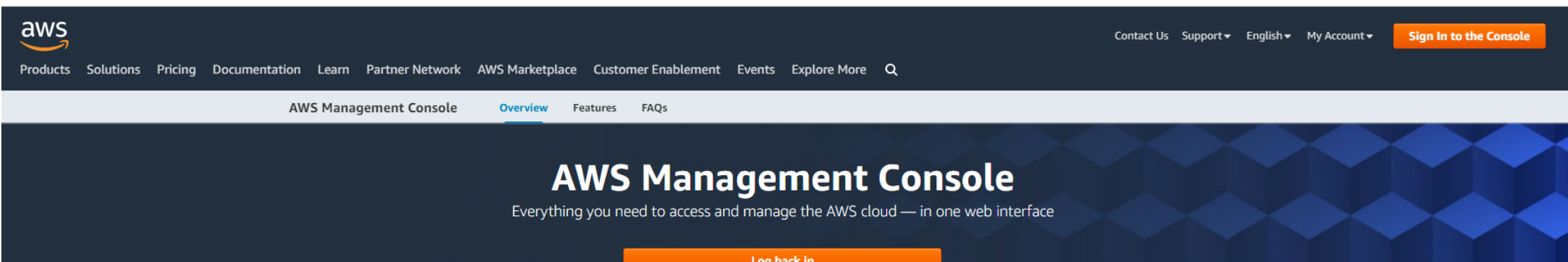
Clock on the 'Add' button

That's it! Your data is being redirected to your Nightscout!

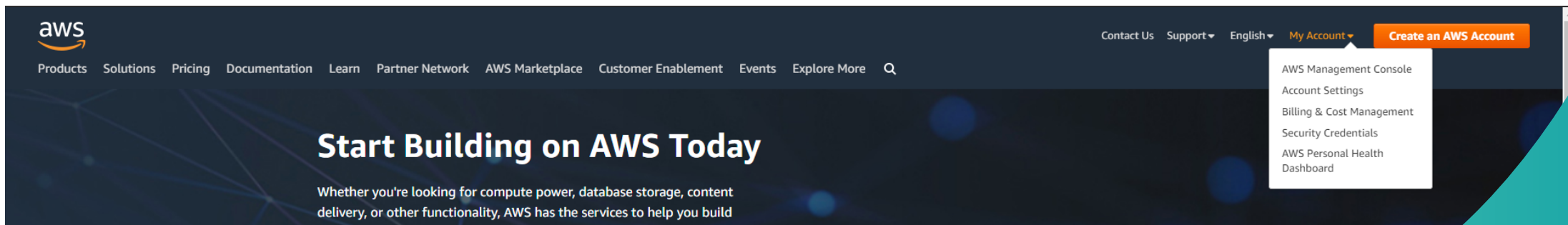


Notes:

How to load the AWS console if you need to go back to it again. One way is to load:



If you enter from <https://aws.amazon.com>, select from the dropdown menu 'My Account' and then 'AWS Management Console'



The given code and manual are provided 'As is'. Use them with care and under your own responsibility. We can't guarantee you technical support by e-mail, message, phone call or any other way of communication due to the busy schedule of the authors. We ask for understanding.